Exploring Reinforcement Learning for Mobile Percussive Collaboration

Nate Derbinsky & Georg Essl

University of Michigan

Our Research Goal

Real-time musical expression on mobile devices

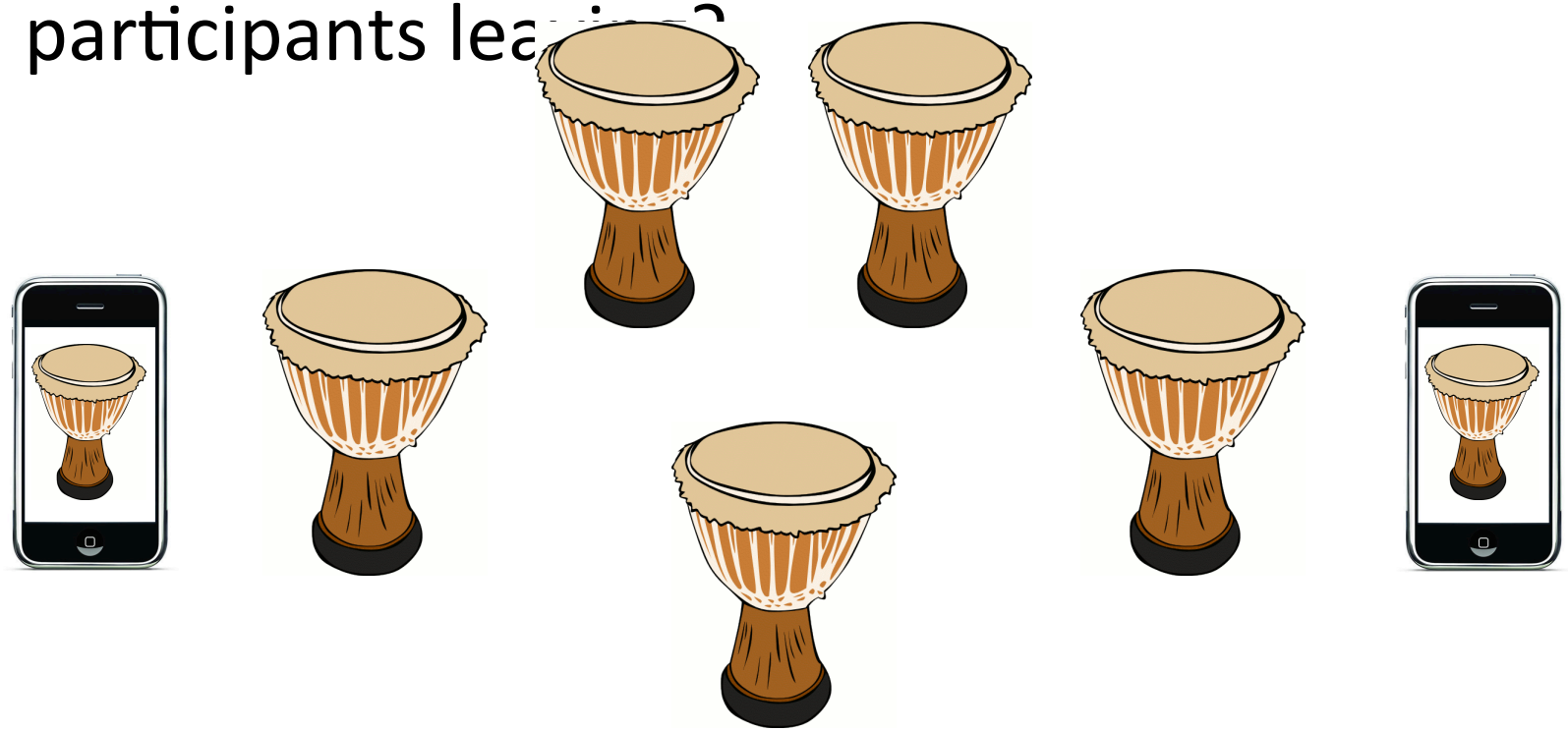
This Paper

Focus. collaborative percussive performance

This Paper

Focus. collaborative percussive performance

- Can reinforcement learning (RL) adapt to participants lea



Outline

Reinforcement Learning

- What is RL?
- Why is it appropriate for mobile percussive collaboration?

System Design

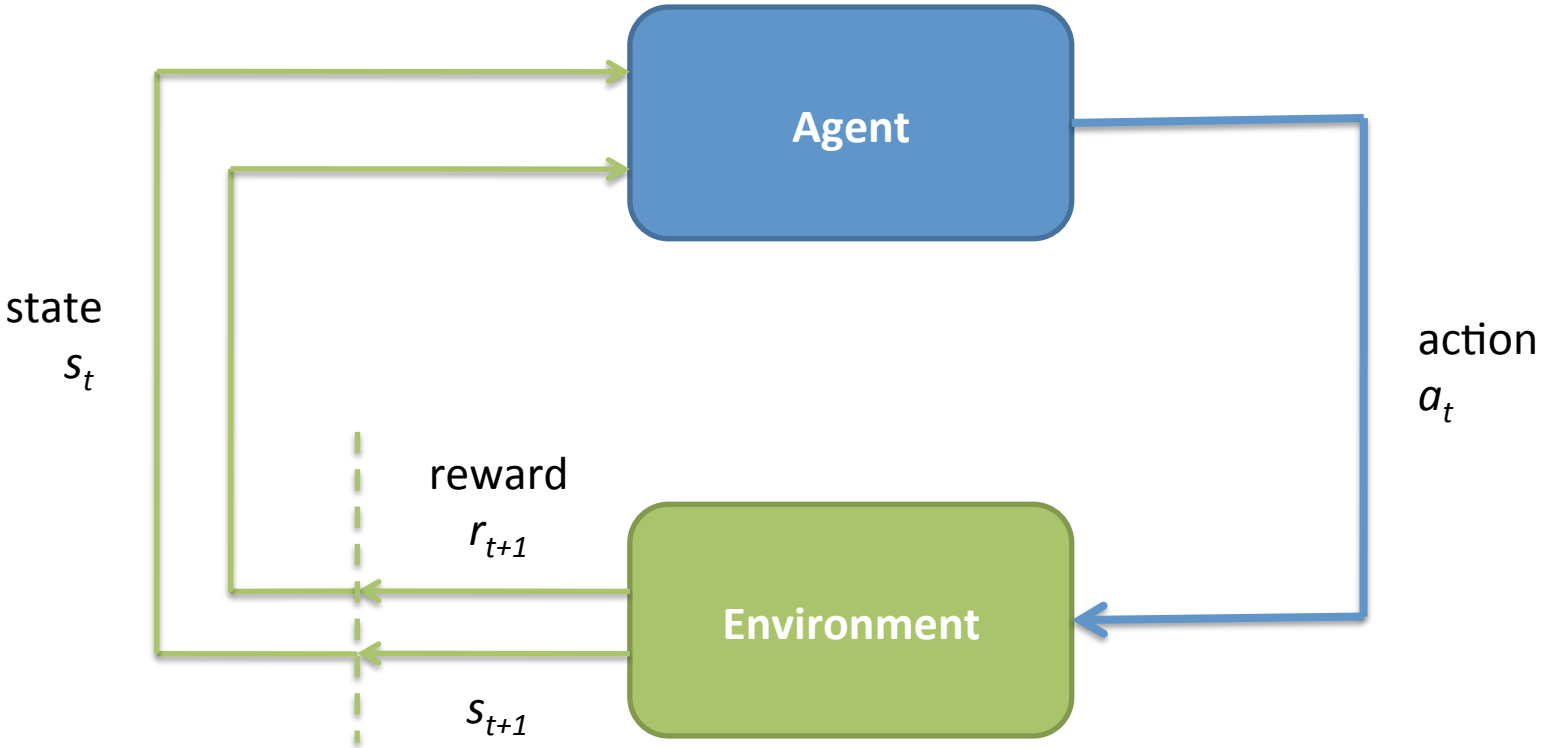
- Learning agent
- Performance interface

Evaluation

- Data sets, methodology, conditions
- Results and analysis

Reinforcement Learning (RL)

Goal: action selection policy such as to maximize expected receipt of future reward



Why RL?

- Online, incremental
- Intuitive mapping
  - State: musical context
  - Actions: beat/no-beat
  - Reward: comparison to human performer

RL Design Space

Feature Selection

What aspects of the situation should inform the next decision?

Reward Signal

How, when, and to what degree to provide performance feedback for prior decisions?

Exploration

How, when, and to what extent should decisions deviate from current “best” decision?

Learning Rate

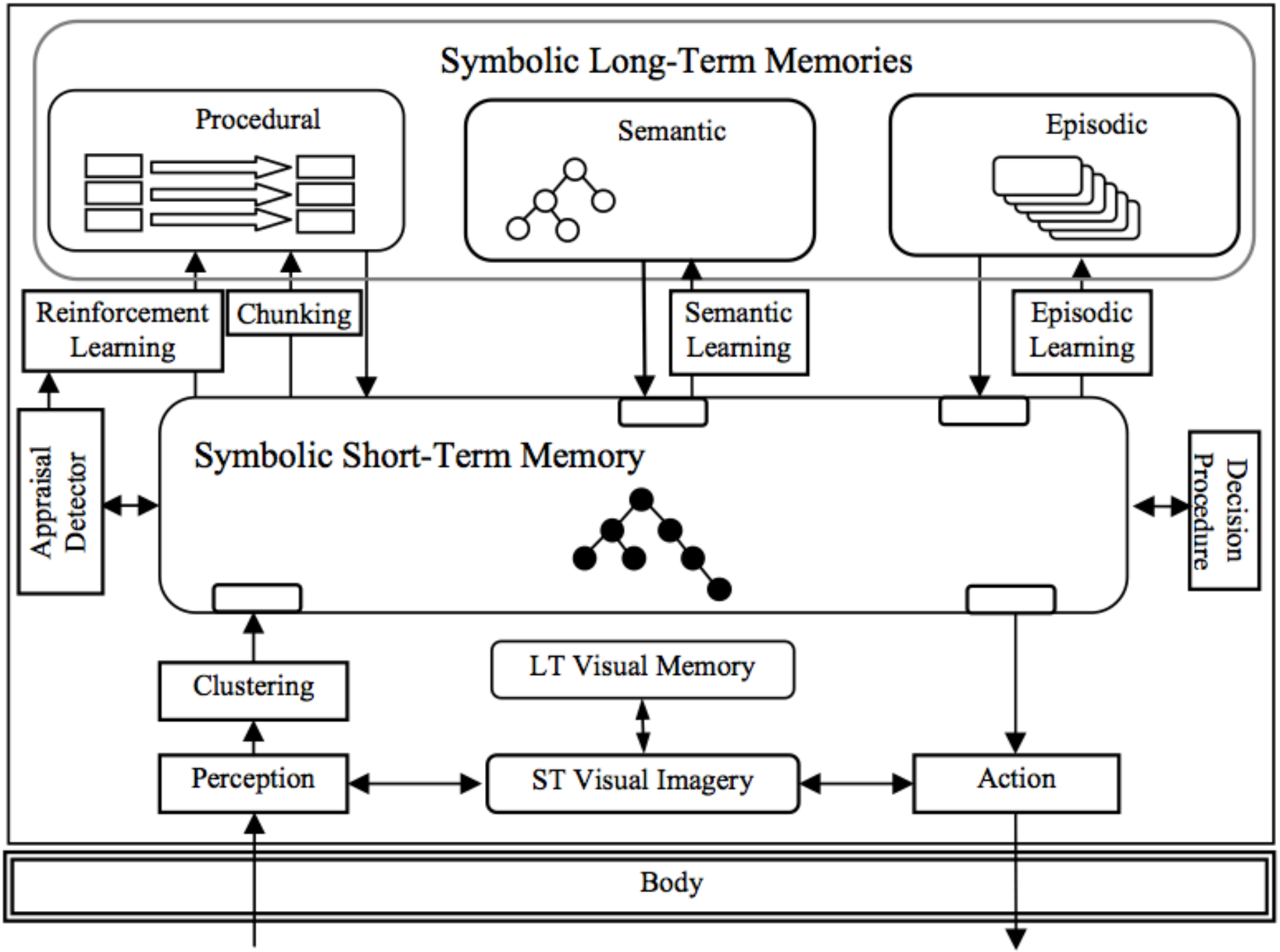
To what extent is the world considered uncertain?

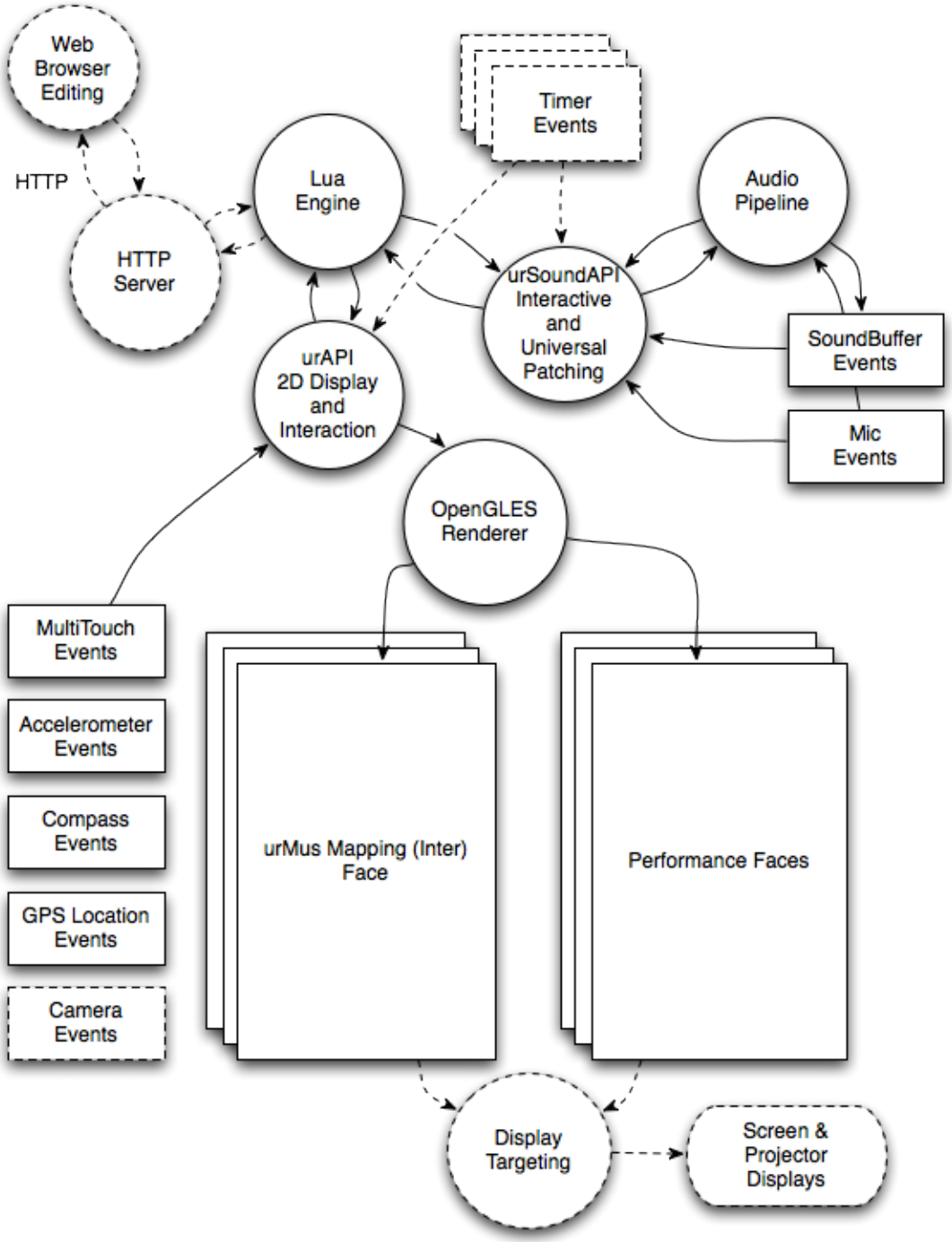
System Desiderata

- Intuitive interface for human performers
  - Includes understanding agent learning
- Learning speed & quality
  - Fast
  - Fidelity, but not “robotic”
- Real-time execution

System Design

- Leverages existing integration between urMus and Soar (NIME '11)
- Components
  - Learning agent
  - Performance UI
  - Learning UI
  - Collaboration UI


Symbolic Long-Term Memories
Procedural
Semantic
Episodic
Reinforcement Learning
Chunking
Semantic Learning
Episodic Learning
Appraisal Detector
Symbolic Short-Term Memory
Decision Procedure
Clustering
LT Visual Memory
Perception
ST Visual Imagery
Action
Body


Web Browser Editing
HTTP
HTTP Server
Lua Engine
Timer Events
Audio Pipeline
urSoundAPI Interactive and Universal Patching
SoundBuffer Events
Mic Events
urAPI 2D Display and Interaction
OpenGLES Renderer
MultiTouch Events
Accelerometer Events
Compass Events
GPS Location Events
Camera Events
urMus Mapping (Inter) Face
Performance Faces
Display Targeting
Screen & Projector Displays

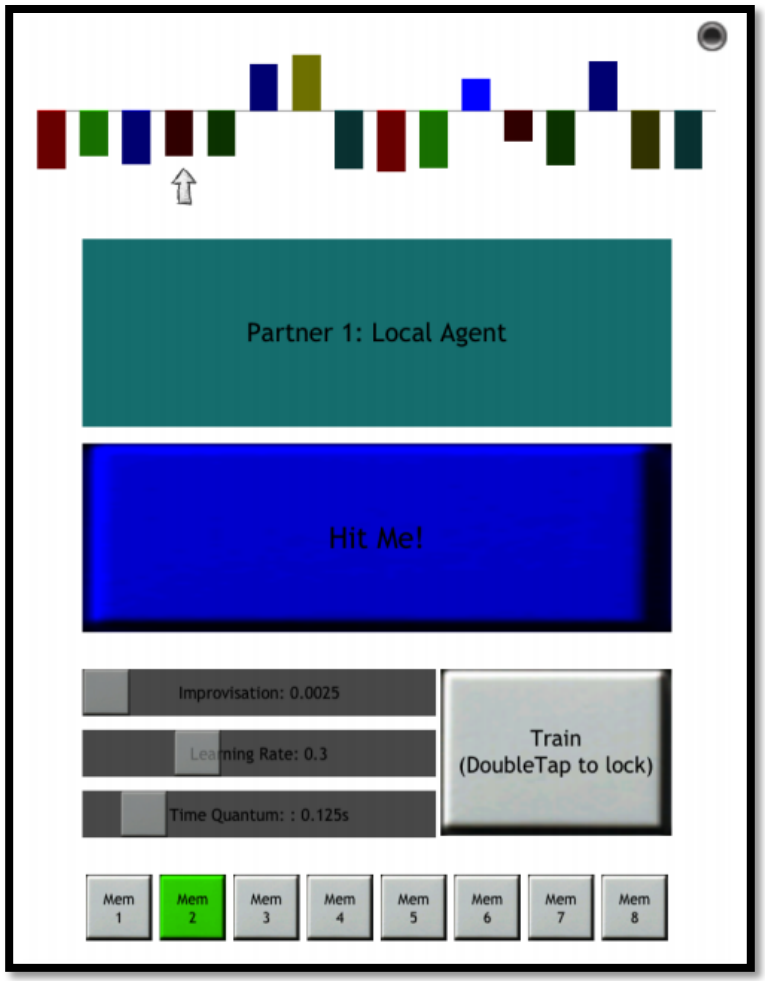
Using Soar in urMus

```
r = Region ()
r:SoarLoadRules (" simon -rl", " soar ")
timeWme = r: SoarCreateConstant (0, " time ", clickcount )
r: SoarExec (" step ".. delayDecisions )
r: SoarDelete ( timeWme )
name , params = r:SoarGetOutput ()
```

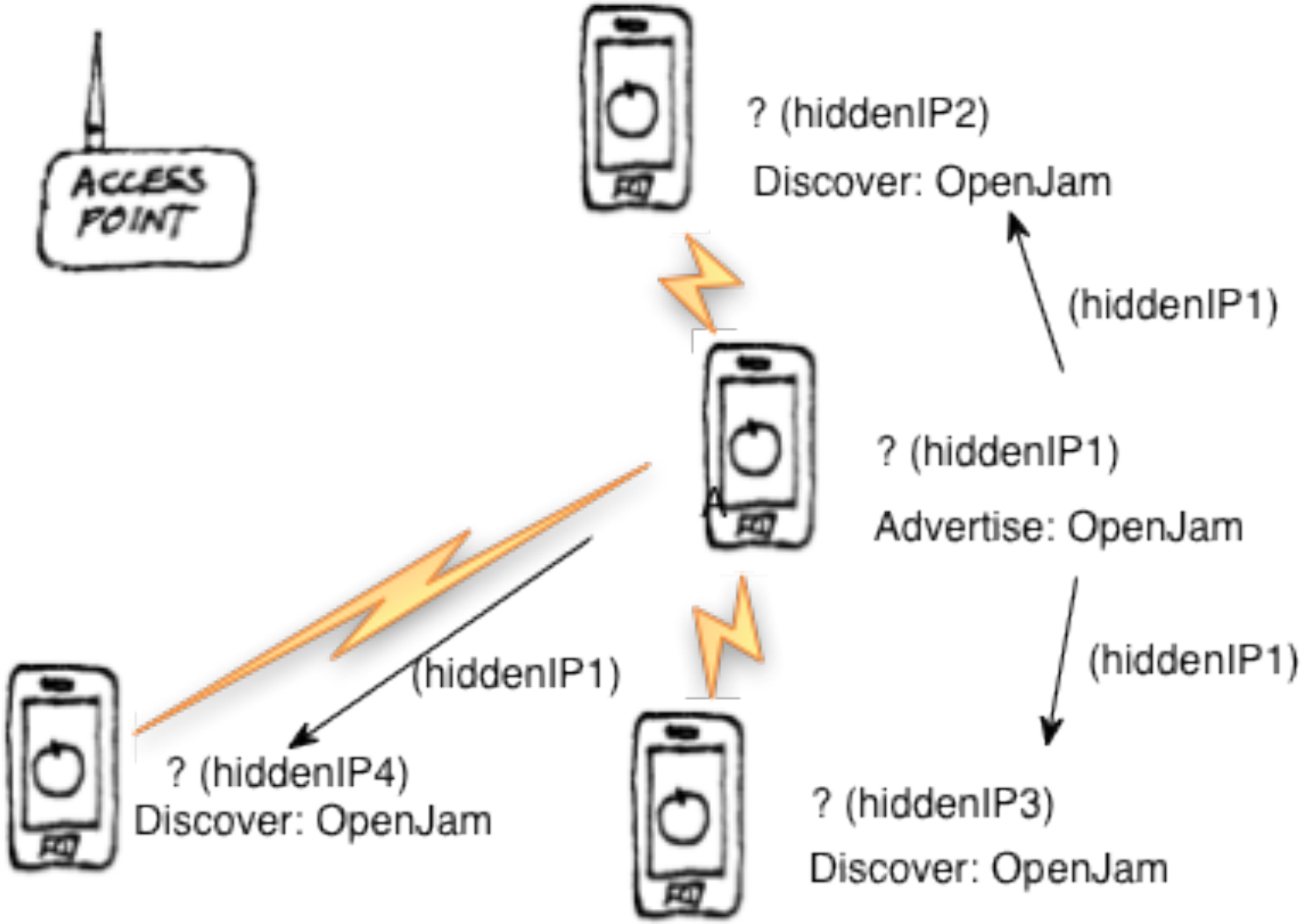
Learning Agent

- Discretized time
  - Each time step is either learning or performing
- SARSA online TD-Learning
  - Reward: +1/-1
  - Discount (γ): 0.9
  - Learning (α): variable
  - Exploration: Boltzmann, variable temperature (τ)

User Interface



- Performance
  - Visual + Auditory
- Learning
  - Visualization
  - Control
- Collaboration
  - Human + Agent
  - Zeroconf Networking


ACCESS POINT
? (hiddenIP2)
Discover: OpenJam
(hiddenIP1)
? (hiddenIP1)
Advertise: OpenJam
(hiddenIP1)
(hiddenIP1)
? (hiddenIP4)
Discover: OpenJam
? (hiddenIP3)
Discover: OpenJam

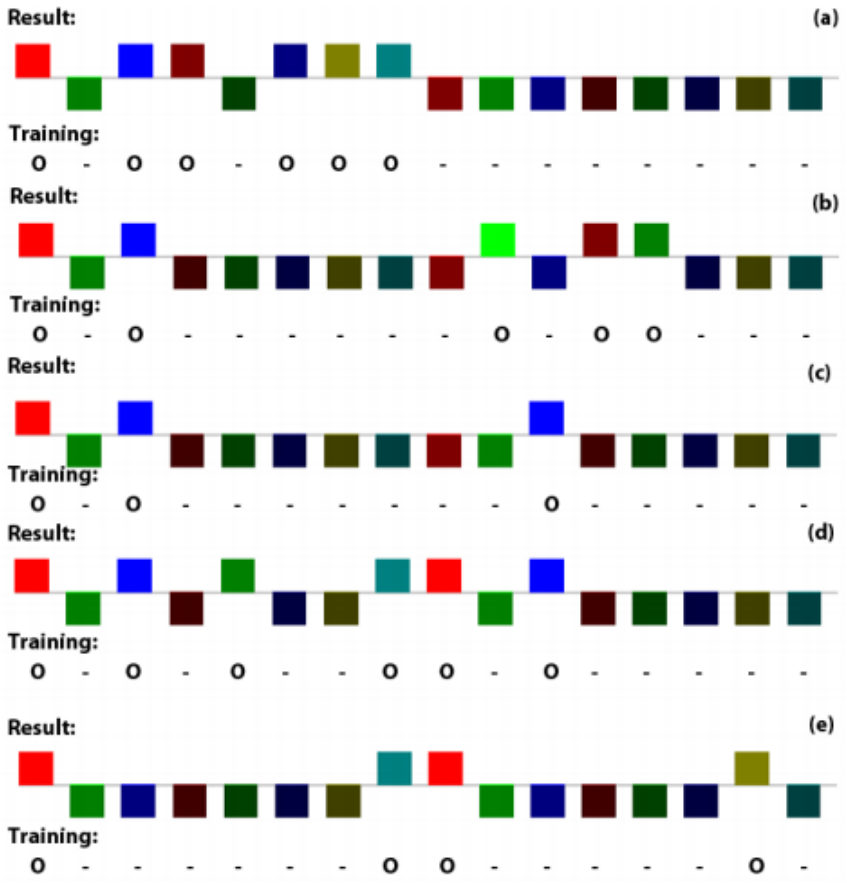
Evaluation

- 5 drum patterns (16b)
- 10 training trials
- 125,000 data points

- Feature representation
  - Absolute time (based upon data*)

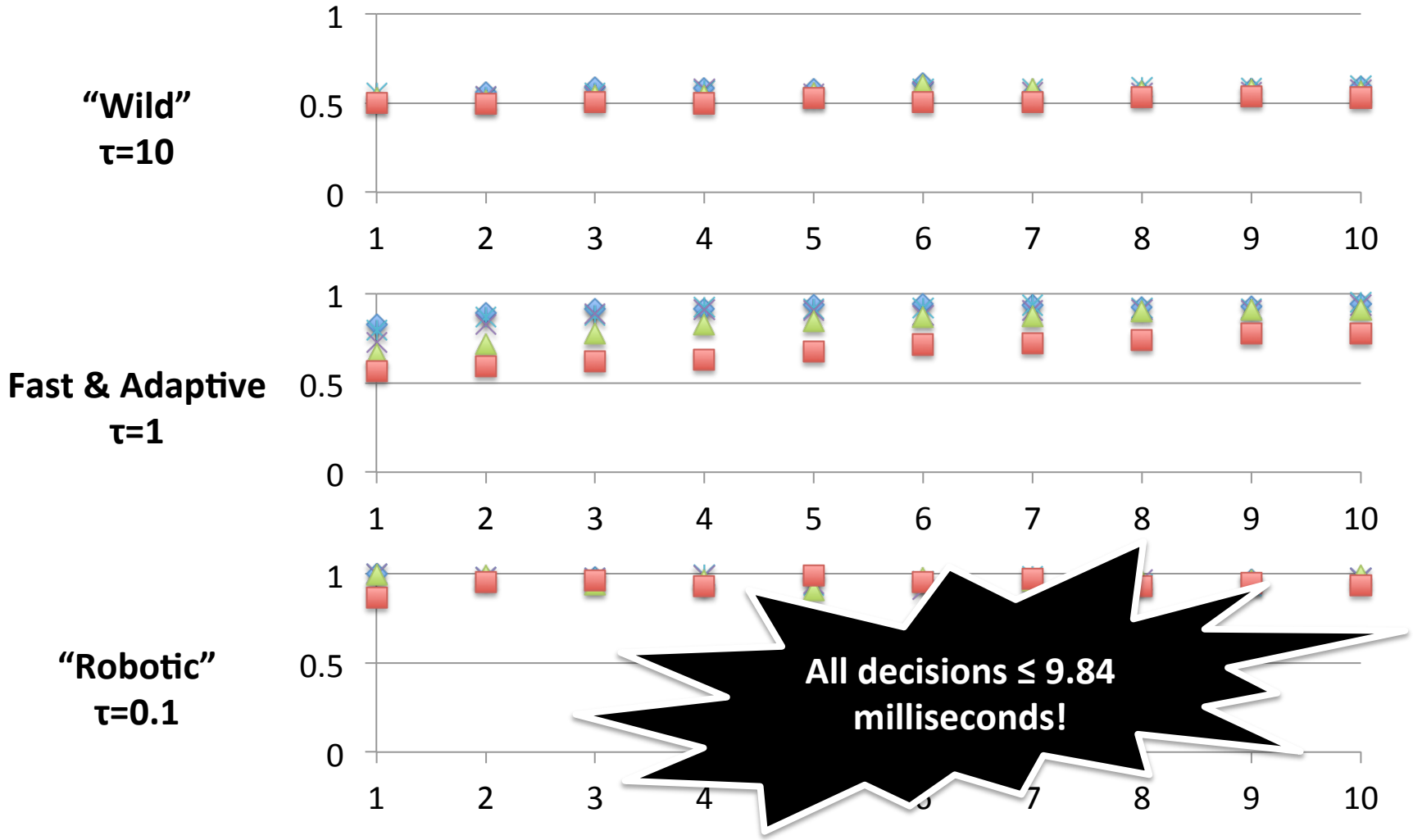
- Noise models
  - None (baseline)
  - Point error (10%, 35%, 25%)
  - Systemic error (10-30%)

- Metrics
  - Accuracy (% match input)
  - Reactivity (max decision)



Learning vs. Improvisation

Avg. Accuracy vs. Trials (by L-rate, α=0.1-0.9)



Summary

Developed and evaluated real-time, collaborative percussive system for mobile devices

- urMus + Soar
- Learning via reinforcement

Explored design space ala musical expression

- Feature representation (context), exploration (improvisation), learning rate (speed), …

Future work

- Expanding rhythmic representation!

Thank You :)

Questions?